**ĐỀ CƯƠNG DỰ ÁN HỌC TẬP: ỨNG DỤNG KẾT NỐI FREELANCER**

**THIẾT KẾ / DỊCH THUẬT / GÕ VĂN BẢN**

**Người thực hiện: Bùi Sơn Tùng, Nguyễn Phú Nam,Đinh Quốc Đạt**  
**Ngày bắt đầu: 27/04/2025**  
**Thời gian thực hiện: 2 tháng**

**I. TỔNG QUAN BÀI TOÁN**

**1. Mô tả bài toán**

Xây dựng một ứng dụng web đơn giản nhằm kết nối giữa:

* **Freelancer**: Người cung cấp dịch vụ thiết kế, dịch thuật, gõ văn bản
* **Khách hàng**: Người có nhu cầu sử dụng các dịch vụ trên

**2. Mục tiêu của ứng dụng**

* Tạo môi trường kết nối cơ bản giữa freelancer và khách hàng
* Cung cấp hệ thống quản lý công việc đơn giản
* Cho phép đánh giá chất lượng dịch vụ
* Triển khai giao diện người dùng thân thiện

**3. Phạm vi bài toán**

* **Loại dịch vụ**: Thiết kế, dịch thuật, gõ văn bản (data entry)
* **Nền tảng**: Web responsive
* **Đối tượng người dùng**: Phục vụ mục đích học tập và chấm điểm

**II. PHÂN TÍCH YÊU CẦU**

**1. Yêu cầu chức năng**

**1.1. Chức năng chung**

* Đăng ký, đăng nhập, quản lý tài khoản
* Tìm kiếm cơ bản dịch vụ/freelancer
* Hệ thống tin nhắn đơn giản

**1.2. Chức năng dành cho Freelancer**

* Tạo và quản lý hồ sơ cá nhân
* Đăng ký kỹ năng và dịch vụ cung cấp
* Đặt giá dịch vụ và thời gian hoàn thành
* Nhận đánh giá từ khách hàng

**1.3. Chức năng dành cho Khách hàng**

* Đăng tin yêu cầu dịch vụ
* Duyệt và lựa chọn freelancer
* Đánh giá dịch vụ sau khi hoàn thành

**1.4. Chức năng dành cho Quản trị viên**

* Quản lý cơ bản người dùng và dự án
* Phê duyệt đăng ký người dùng

**2. Yêu cầu phi chức năng**

* **Bảo mật**: Bảo vệ thông tin cá nhân cơ bản
* **Trải nghiệm người dùng**: Giao diện đơn giản, dễ sử dụng
* **Triển khai**: Dễ dàng triển khai và demo cho mục đích học tập

**III. THIẾT KẾ HỆ THỐNG**

**1. Kiến trúc tổng thể**

* **Kiến trúc 3 tầng**:
  + Tầng trình bày (Presentation Layer)
  + Tầng nghiệp vụ (Business Layer)
  + Tầng dữ liệu (Data Layer)
* **Mô hình Client-Server**
* **RESTful API** cho giao tiếp giữa Frontend và Backend

**2. Thiết kế cơ sở dữ liệu**

**2.1. Sơ đồ quan hệ thực thể (ERD)**

Các thực thể chính:

* **Users**: Thông tin người dùng
* **Freelancers**: Thông tin chi tiết về freelancer
* **Clients**: Thông tin chi tiết về khách hàng
* **Services**: Dịch vụ được cung cấp
* **Projects**: Thông tin dự án
* **Reviews**: Đánh giá từ khách hàng
* **Messages**: Tin nhắn giữa người dùng
* **Categories**: Danh mục dịch vụ

**2.2. Cấu trúc bảng dữ liệu (Thu gọn)**

**Users**

* UserID (PK)
* Username
* Email
* Password (đã mã hóa)
* FullName
* Avatar
* UserType (Freelancer/Client/Admin)
* Status
* CreatedAt

**Freelancers**

* FreelancerID (PK)
* UserID (FK)
* Bio
* Skills
* HourlyRate
* Rating

**Clients**

* ClientID (PK)
* UserID (FK)
* CompanyName

**Services**

* ServiceID (PK)
* CategoryID (FK)
* FreelancerID (FK)
* Title
* Description
* Price
* DeliveryTime
* Status

**Projects**

* ProjectID (PK)
* ClientID (FK)
* Title
* Description
* CategoryID (FK)
* Budget
* Deadline
* Status
* PostedAt

**Reviews**

* ReviewID (PK)
* ProjectID (FK)
* ReviewerID (FK)
* ReviewedID (FK)
* Rating
* Comment
* CreatedAt

**Messages**

* MessageID (PK)
* SenderID (FK)
* ReceiverID (FK)
* Content
* SentAt

**Categories**

* CategoryID (PK)
* Name
* Description

**3. Thiết kế giao diện người dùng**

**3.1. Trang chủ**

* Banner quảng cáo và giới thiệu
* Danh mục dịch vụ
* Freelancer nổi bật
* Dự án mới đăng

**3.2. Trang cá nhân freelancer**

* Thông tin cá nhân và kỹ năng
* Portfolio
* Đánh giá từ khách hàng
* Dịch vụ đang cung cấp

**3.3. Trang chi tiết dự án**

* Mô tả dự án
* Yêu cầu và ngân sách
* Danh sách đề xuất
* Trạng thái dự án

**3.4. Dashboard cho người dùng**

* Tổng quan dự án/công việc
* Thông báo
* Tin nhắn
* Giao dịch gần đây

**IV. LỰA CHỌN CÔNG NGHỆ**

**1. Công nghệ Backend**

* **Ngôn ngữ**: Node.js với Express.js
* **API**: RESTful API với JSON
* **Xác thực**: JWT (JSON Web Token)
* **Xử lý tệp**: Lưu trữ local hoặc Cloudinary (miễn phí)

**2. Công nghệ Frontend**

* **Framework**: React.js
* **UI Library**: Bootstrap
* **API Client**: Axios

**3. Cơ sở dữ liệu**

* **RDBMS**: MySQL
* **Hosting**: XAMPP (local) hoặc hosting miễn phí

**4. Công cụ phát triển**

* **Version Control**: Git và GitHub
* **API Documentation**: Postman
* **Testing**: Jest (cơ bản)

**V. PHÂN CÔNG CÔNG VIỆC**

**1. Thành viên 1: Phân tích thiết kế và cơ sở dữ liệu**

* Phân tích yêu cầu cơ bản
* Thiết kế và triển khai cơ sở dữ liệu
* Xây dựng mô hình dữ liệu và quan hệ

**Công việc cụ thể**:

1. Tạo Use Case Diagram đơn giản
2. Thiết kế ERD và triển khai cơ sở dữ liệu MySQL
3. Tạo dữ liệu mẫu cho demo
4. Viết tài liệu mô tả cơ sở dữ liệu

**2. Thành viên 2: Xử lý tương tác (Frontend và sử dụng API)**

* Thiết kế và phát triển giao diện người dùng
* Kết nối với API Backend
* Thiết kế responsive đơn giản

**Công việc cụ thể**:

1. Thiết kế wireframe cơ bản
2. Xây dựng các trang chính (trang chủ, đăng ký, đăng nhập, profile)
3. Phát triển trang quản lý dự án và dịch vụ
4. Triển khai chức năng tìm kiếm đơn giản

**3. Thành viên 3: Xử lý logic nghiệp vụ ứng dụng (API Backend)**

* Phát triển RESTful API cơ bản
* Triển khai hệ thống xác thực đơn giản
* Xử lý logic nghiệp vụ chính

**Công việc cụ thể**:

1. Xây dựng các endpoints API cơ bản
2. Triển khai hệ thống authentication với JWT
3. Xử lý logic cho quản lý người dùng, dự án và dịch vụ
4. Viết các chức năng xử lý dữ liệu và truy vấn

**VI. KẾ HOẠCH TRIỂN KHAI (2 THÁNG)**

**1. Giai đoạn 1: Khởi động và phân tích (1 tuần)**

* Phân tích yêu cầu cơ bản
* Lựa chọn công nghệ và công cụ
* Thiết lập môi trường phát triển

**2. Giai đoạn 2: Thiết kế và phát triển cơ sở (2 tuần)**

* Thiết kế cơ sở dữ liệu
* Phát triển khung API cơ bản
* Xây dựng khung giao diện người dùng
* Triển khai hệ thống xác thực đơn giản

**3. Giai đoạn 3: Phát triển tính năng chính (3 tuần)**

* Phát triển chức năng đăng tin và tìm kiếm
* Xây dựng chức năng quản lý dự án
* Triển khai chức năng tin nhắn đơn giản
* Phát triển chức năng đánh giá

**4. Giai đoạn 4: Hoàn thiện và kiểm thử (2 tuần)**

* Kiểm thử chức năng
* Sửa lỗi và cải thiện
* Hoàn thiện UI/UX
* Chuẩn bị tài liệu và demo

**VII. RỦI RO VÀ GIẢI PHÁP**

**1. Rủi ro kỹ thuật**

* **Thiếu kinh nghiệm với công nghệ**: Tập trung vào học hỏi và nghiên cứu trước khi triển khai
* **Lỗi tích hợp giữa Frontend và Backend**: Xây dựng API documentation rõ ràng và thống nhất cách tương tác
* **Vấn đề xác thực và bảo mật**: Áp dụng các biện pháp bảo mật cơ bản, xác thực JWT

**2. Rủi ro quản lý**

* **Deadline không thực tế**: Chia nhỏ công việc, tập trung vào các chức năng cốt lõi
* **Thay đổi yêu cầu**: Xác định rõ phạm vi từ đầu, hạn chế thay đổi
* **Phối hợp giữa các thành viên**: Họp nhóm thường xuyên, sử dụng công cụ quản lý mã nguồn

**VIII. KẾT LUẬN**

Ứng dụng kết nối freelancer và người có nhu cầu thiết kế/dịch thuật/gõ văn bản hứa hẹn mang lại giá trị lớn cho cả người cung cấp dịch vụ và khách hàng. Với kiến trúc được thiết kế tốt và sử dụng các công nghệ hiện đại, ứng dụng có thể xây dựng một nền tảng ổn định, linh hoạt và có khả năng mở rộng.

Sự phân công rõ ràng giữa ba thành viên nhóm về phân tích thiết kế, frontend và backend sẽ đảm bảo quá trình phát triển diễn ra hiệu quả và đồng bộ. Kế hoạch triển khai chi tiết cùng với việc xác định trước các rủi ro tiềm ẩn và giải pháp sẽ giúp dự án được hoàn thành đúng thời hạn và đạt chất lượng tốt.